Monitoring Compute Health

David Tucker
TECHNICAL ARCHITECT & CTO CONSULTANT
@_davidtucker_ davidtucker.net

App Service

App Service for Containers

Virtual Machines

Azure Kubernetes Service (AKS)

Container Instances (ACI)

Service Fabric

Azure Functions

Azure Batch

Azure Compute Services

App Service Health Checks

Now generally available (as of August 2020)

App Service will handle routing only to health instances for your application

Path must be provided where health will be verified

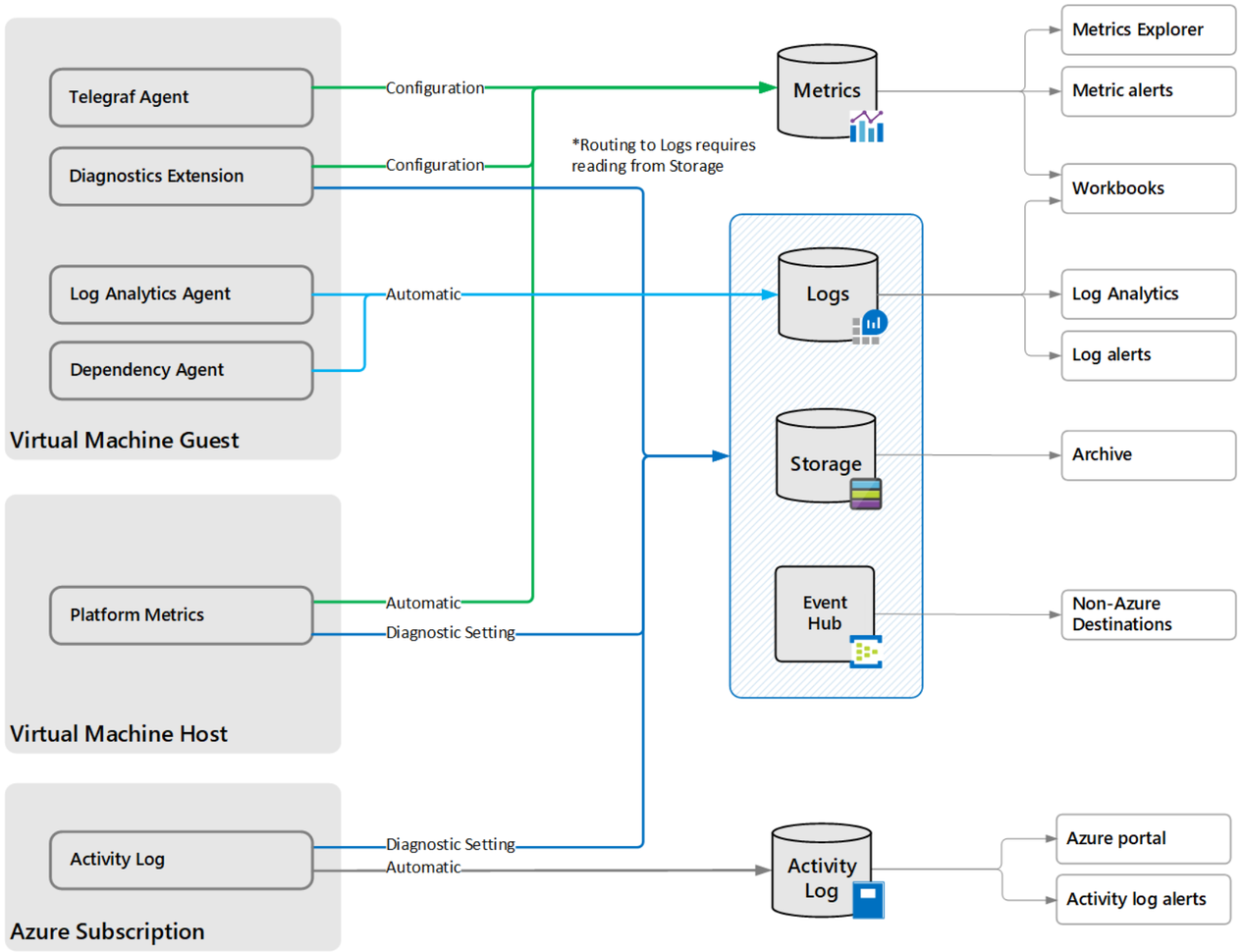
For an app to be healthy, it must:

- Return within one minute
- Return a status code of 200 and 299

Unhealthy apps should return a 5xx response

App Service will not follow 302 redirects on the health check path

Azure Virtual Machine Monitoring



Health Monitoring Approaches

Azure Monitor

System Center Operations Manager

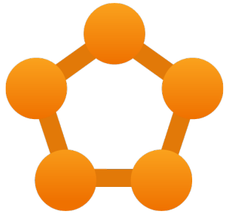
System Center Operations Manager with Azure Monitor

Azure Batch

At its core, Azure Batch runs virtual machines to tackle your batch workloads. You can utilize the same monitoring and alerting capabilities with Azure Batch as you would with your other virtual machines.

“The **Service Fabric** platform includes a health model, which provides extensible health reporting for the status of entities in a cluster. Each node, application, service, partition, replica, or instance, has a continuously updatable health status.”

Microsoft Azure Documentation



Service Fabric Health Checks

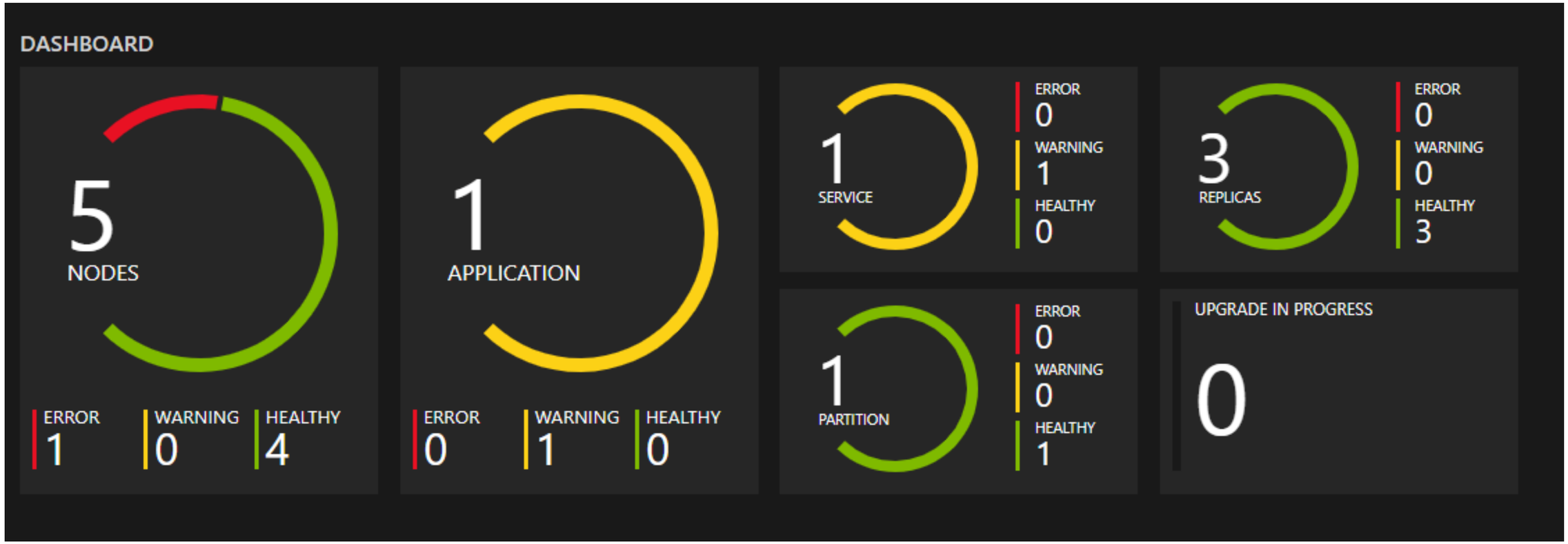
Health check approach is designed to support zero downtime upgrades

Supports health checking in a hierarchy from clusters all the way down to replicas

Each entity produces a health report to indicate its state to the health store

Each entity has configurable aspects to its specific health report

Service Fabric Dashboard



Function App Health Checks

The serverless nature means that many aspects don’t need to be tested

Health checks can still be created to test dependent systems

Alerts can be tied to these health checks

Warm analysis is possible using the application map functionality if alerts arise

Container Health Checks

Azure Container Services

Web App for Containers (App Service)

Azure Container Instances (ACI)

Azure Kubernetes Service (AKS)

Web App for Containers

Configuration of a health check for an App Service container works just like configuring a health check for other application types.

Azure Container Instances

Health checks for containers are created as probes with a configurable check period

ACI supports two probe types: liveness and readiness

Probes support executing a command on the container or performing a GET HTTP request

Probes are configured in the container groups definition YAML file

Container Instances Probes

Liveness Probe

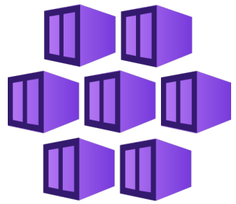
A probe designed to verify that a container is healthy - could result in a container shutdown

Readiness Probe

A probe designed to see if a container is temporarily unable to process requests

Probe Configuration

```
apiVersion: 2019-12-01
location: eastus
name: ps-container-liveness-1
properties:
  containers:
  - name: nodeserver
      livenessProbe:
        exec:
            command:
                - "cat"
                - "/tmp/healthy"
        periodSeconds: 5
  osType: Linux
  restartPolicy: Always
type: Microsoft.ContainerInstance/containerGroups
```



Azure Kubernetes Service

The approach for health checks in ACI is based on Kubernetes, so they are similar

Kubernetes supports three probe types:

- Liveness
- Readiness
- Startup

Kubernetes provides the following probe approaches:

- Exec
- HTTP GET
- TCP

Configuring Readiness and Liveness Probes

Demo

Implementing a liveness probe for ACI

Implementing a readiness probe for ACI

Next Steps

Next Steps

- Review material presented in this course
- Configure your own health checks across your applications
- Continue in the learning path
- Register for your AZ-400 certification exam